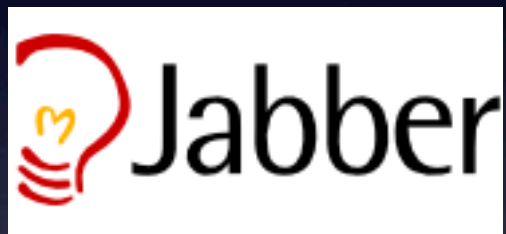


peter saint-andre

@stpeter



# #realtimeconf

presence

first, a short history  
of messaging...

RTT  
London  $\leq$  Calcutta

1800: 2 years  
(sailing ship)



1914: 1 month  
(steamship)

1950: 1 week  
(airmail)

1980: 2 days  
(overnight mail)

1993: ~10 minutes  
(email)

yes, the dark ages

but if you replied really  
fast...

it almost seemed...

like the other person...



*was there*

thus the idea of  
presence was born

presence was a key to  
“instant” messaging

# I 999: sub-second RTTs

IM is the original  
realtime technology  
(for messaging)

e.g., XMPP = Extensible  
Messaging *and* Presence  
Protocol

we've been keeping it  
realtime since 1999 ;-)

what have we  
jabberites learned?



realtime is *hard*

# memory matters

lots of presence state in  
an IM server

so: keep session objects  
small

# bandwidth matters

90% of traffic in an IM  
system is presence

so: keep notifications  
small

# recency matters



stale data is almost  
worse than no data

so: push updates out  
fast

also: detect unavailable  
endpoints

privacy matters

not everyone has a right  
to see my presence

so: allow access only  
with explicit approval

(in XMPP, tied to your  
buddy list)

# specialization matters



presence is  
a pubsub mechanism

but presence is not a  
*generic* pubsub  
mechanism

(in XMPP, we have one  
of those, too ;-)

so: don't push every kind  
of data via presence

presence is availability,  
attention, focus,  
capability, etc.

# security matters

SSL/TLS everywhere  
is only the beginning

end-to-end encryption  
isn't easy (we've failed  
~4 times)



# internationalization matters

Unicode / UTF-8  
is only the beginning

here we've semi-failed  
only once – so far ;-)

# federation matters

single-service silos  
provide a single point of  
failure/attack/control

decentralized tech is  
harder but it's the  
Right Thing™ to do

that's nice, but...

presence so 90s



is it still relevant?

true, it's not critical for  
“neartime” apps

(Skype has presence,  
Twitter doesn't)

but it's key for  
“unsocial” apps like  
IM and collaboration

trigger for  
spontaneous / intensive  
interaction

voice, video, sharing,  
collaborative editing,  
whiteboarding, etc.

i.e., all that realtime stuff

so presence is a stable,  
well-known technology



but there are still  
opportunities here...

presence is often tied to  
a buddy list, but it's not  
limited to people

so: presence about bots,  
components, services,  
devices, etc.

(e.g., use in smart grid /  
Internet of Things)

(e.g., component  
presence inside a  
cluster)

challenging to manage  
multiple presence  
endpoints

so: better client-side  
composition and  
visualization

also: presence  
aggregation services



(like, say, superfeedr or  
fanout.io for presence)

build on presence for  
extended services

so: augment with  
device capabilities

also: extended presence  
(geoloc, activity, mood,  
ambient information)

more interactions now  
are in mobile devices  
and web browsers

so: more and better  
presence in mobile apps

also: per-app presence  
in browsers

also: XMPP over  
WebSocket



but don't be scared by  
the X in XMPP

angle-brackets  
are not required

it's easy to use  
web APIs like strophe.js

too many people are still  
working synchronously

(e.g., sending slideware  
over email)

so: build more  
interactive ways to work  
together in real time

rethink existing  
collaboration patterns

presence happens in a  
specific place



(e.g., in the browser  
at a webapp)

web presence + webrtc  
= realtime goodness

you can even  
“advertise your app”  
via presence status...

```
<presence from='stpeter@jabber.org'>  
  <show>dnd</show>  
  <status>in a PiratePad session</status>  
</presence>
```

lots of presence at large  
silos, but your webapp  
has presence too

please, don't be evil  
a small silo

use XMPP to share  
presence across  
domains

federate or die



presence is the dialtone  
for the realtime internet

presence is a catalyst for  
and byproduct of  
realtime interaction

presence is a solved  
problem — just use it!